Atty Docket No.: JGR 1006-2

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop RCE: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on 27 March 2006

Tracey Horne

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Bart A. MELTZER et al. Application No. 09/633,365 **Confirmation No. 3951** Filed: 07 August 2000 Title: **Registry for Trading Partners Using Documents for Commerce in Trading Partner Networks**	Group Art Unit: 2141 Examiner: Kenneth R. COULTER **CUSTOMER NO. 22470**

MAIL STOP RCE
Commissioner for Patents
Alexandria, VA 22313-1450

Declaration under 37 CFR § 1.132 of Jay M. Tenenbaum, Ph.D.

I, Jay M. Tenenbaum, declare as follows:

1. I am Chairman and co-founder of CommerceNet and a named author of two articles discussed in this declaration. All of the statements made herein are my own personal knowledge or of my own opinion, based on my training and experience, except where stated on information and belief. I could competently testify thereto if called as a witness.

2. This declaration is given in support of the application entitled "Registry for Trading Partners Using Documents for Commerce in Trading Partner Networks," U.S. patent application serial number 09/633,365, filed on August 7, 2000. In advance of preparation of this declaration, I looked at the Tenenbaum and Sriram articles discussed below and was provided with copies of the parent 6,125,391 patent, applicants' Response to Office Action mailed on March 8, 2005 and the Examiner's Final Office

Action mailed September 26, 2005. During preparation of this declaration, I further reviewed a 1999 article that is identified below.

3. A version of my resume is attached. It is reasonably current, but has not been updated or double checked for this declaration.

4. One of my current positions is as Chairman of CommerceNet (at www.commerce.net). CommerceNet was founded in 1994. It provided a loan to a predecessor of Veo, to help it get started, and received stock as a result. Veo merged into Commerce One, where I served as Chief Scientist. At Veo, I was involved in hiring at least some of the inventors listed on this application, including Bart Meltzer and Robert Glushko. At CommerceNet, I have ongoing contact with some of the inventors listed, including Andrew Davidson.

5. I have been interested in the disposition of Commerce One's patents and patent applications, because they are very significant to business-to-business e-commerce. CommerceNet made an effort to organize several companies to collectively buy the Commerce One portfolio at a bankruptcy auction in late 2004. There are various public reports of this effort.

6. I am informed and believe that the Commerce One patents and patent applications of interest that were auctioned by a bankruptcy court are now assigned to Open Invention Network ("OIN"), an organization formed by IBM, Phillips, Sony, RedHat and Novell, which has the web site www.openinventionnetwork.com. I am not involved in OIN, but I appreciate their stated dedication to acquire patents and offer them royalty-free to promote Linux and spur innovation globally.

7. I was working in the field of e-commerce architecture when my article, Tenenbaum, Jay M., Tripatinder S. Chowdhry and Kevin Hughes, "Eco System: An Internet Commerce Architecture" Computer May 1997: 48-55 ("Eco System article", attached) was written and subsequently when Veo-related companies began work on a document-based interface to loosely coupled services. I am familiar with the ordinary level of skill in e-commerce architecture that was common among software engineers in 1997-98, because I trained, worked with and observed the work of many software engineers in that era. For instance, around that time, I taught e-commerce and computer science at Stanford and asked some of my students to try and build e-commerce services using CORBA.

8. The Eco System article does not teach a software engineer of ordinary skill, circa 1997-98, to use a document-based interface to loosely couple services. Historically, Eco's notion of using a document-based interface to e-commerce web services arose after this article was written, from people who I subsequently hired, including Robert Glushko and Bart Meltzer. Hiring Glushko after the article was written changed my thinking, becuase Gushko came out of the document world and had a much different approach. At the time this article was written, I did not have a document-based interface in mind, but rather an interface based on application program interfaces, a la CORBA.

9. The Eco System Article does not describe or suggest a machine readable specification to a business service that includes definitions of or references to definitions of documents to be exchanged with the business service. One familiar with CORBA will understand that the interface to a CORBA process was not, in 1997-98, document-based. The CORBA objects carried by a CORBA bus were not documents.

10. The Eco System Article does not describe or suggest a machine readable specification to a business service that includes definitions of documents to be exchanged with the business service that include markup data to define storage units and logical structures of input and output documents. One familiar with CORBA will understand that the interface to a CORBA process did not, in 1997-98, use markup data to define storage units and logical structures of input and output documents. The CORBA objects carried by the CORBA bus were not documents.

11. Use of documents to define a service interface, instead of CORBA objects, was very innovative because it dramatically simplified the integration of business services and interfaces. A CORBA system, as described in the article, would never have worked as described, because it is too hard to get parties to agree on service APIs, whereas business documents were fairly well standardized (e.g., purchase orders and invoices.)

As a result of developing a document-based interface system, in a matter of a few years, Commerce One had enormous success bringing many businesses together into marketplaces based on loosely coupled services with document interfaces. Commerce One brought together businesses that did not typically cooperate, such as the automotive Big Three in one marketplace and major aircraft manufacturers in

another marketplace. In my opinion, these kinds of marketplaces would not have been commercially attractive if they were designed using CORBA objects.

12. Veo designed software to use document-based interfaces as an improvement on and change from CORBA-object architectures. Veo and Commerce One intentionally developed a non-CORBA software architecture, because of the limitations of CORBA.

13. I have been asked to review and comment on several passages from my article. The passages and my comments follow.

14. On page 48 of the article, the text appears:

> The Internet is revolutionizing commerce. It provides the first affordable and secure way to link people and computers spontaneously across organizational boundaries. This is spawning numerous innovative enterprises—virtual companies, markets, and trading communities.

In my opinion, this passage does not describe or suggest a machine readable specification to a business service that includes definitions of or references to definitions of **documents** to be exchanged with the business service. It does not describe or suggest a machine readable specification to a business service that includes definitions of **documents** to be exchanged with the business service that include **markup data** to define storage units and logical structures of input and output documents. Reference in this passage to "trading communities" does not teach or suggest maintaining a registry of document-based interfaces to web services.

15. On page 51, the text appears:

> *Matchmaking* is a trading post where buyers and sellers can exchange goods or services. This service matches buyers and sellers on the basis of product descriptions and personal or company profiles (like, for example, Sun's Matchmaker).

I do not recall Sun's Matchmaker at this time. As I read this text, it does not describe or suggest a machine readable specification to a business service that includes definitions of or references to definitions of **documents** to be exchanged with the business service.

It does not describe or suggest to me a machine readable specification to a business service that includes definitions of **documents** to be exchanged with the business service that include **markup data** to define storage units and logical structures of input and output documents. Reference in this passage to matching "buyers and sellers on the basis of product descriptions and personal or company profiles" does not teach or suggest maintaining a registry of document-based interfaces to web services.

16. On page 52, the text appears:

> *Scaleable, interchangeable building blocks.* Agents can direct CBL commands to a business, several businesses that have linked their catalogs or processes, a market (comprised of many companies), or a third-party intermediary.

In my opinion, this passage does not describe or suggest a machine readable specification to a business service that includes definitions of or references to definitions of **documents** to be exchanged with the business service. It does not describe or suggest a machine readable specification to a business service that includes definitions of **documents** to be exchanged with the business service that include **markup data** to define storage units and logical structures of input and output documents. Reference in this passage to "several business that have linked their catalogs or processes" does not teach or suggest maintaining a registry of document-based interfaces to web services. Linking of processes, in the context of this article, suggests to me CORBA-styled tight coupling of business processes. Use of CORBA objects to link processes was very different from using document-based interfaces to loosely couple services.

17. I also have been asked to review the side-bar in my article, Sriram, Ram, "AIMSNet", which appears on page 54. My recollection is that AIMSNet was a DARPA research prototype, focused more on collatorative engineering than e-commerce. It was ever commercially deployed. It was based on arcane standards of the DARPA AI knowledge sharing community and had nothing to do with document-based technologies.. Commerce One's success introducing document-based technology to the aeronautical industry far eclipsed AIMSNet. While I do not presently recall the details of AIMSNet and the details are not explained in the side-bar on page 54, I have great certainty that AIMSNet of 1996-97 did not use a registry including document-based interface definitions.

18. One passage of Siram reads:

In my opinion, this passage does not describe or suggest a machine readable specification to a business service that includes definitions of or references to definitions of **documents** to be exchanged with the business service. It does not describe or suggest a machine readable specification to a business service that includes definitions of **documents** to be exchanged with the business service that include **markup data** to define storage units and logical structures of input and output documents. Reference in this passage to "exchange [of] technical and business information" does not teach or suggest maintaining a registry of document-based interfaces to web services.

19. Another passage of Siram reads:

> AIMSNet, an industrial commerce infrastructure, is currently piloted as an aerospace I-market but can be easily customized to several other I-markets including automotive, electronics, and construction.

In my opinion, this passage does not describe or suggest a machine readable specification to a business service that includes definitions of or references to definitions of **documents** to be exchanged with the business service. It does not describe or suggest a machine readable specification to a business service that includes definitions of **documents** to be exchanged with the business service that include **markup data** to define storage units and logical structures of input and output documents. Reference in this passage to "an aerospace I-market" does not teach or suggest maintaining a registry of document-based interfaces to web services.

20. Attached to this declaration is a copy of the article, Glushko, Robert J., Jay M. Tenenbaum, Bart Meltzer, "An XML Framework for Agent-based E-commerce" Communications of the ACM, Vol. 42, No. 3, pp.106-109 & 111-114 (March 1999). I have reviewed this article, with particular attention to pp. 108, 109 & 111. My co-authors, Glushko and Meltzer are inventors named on this application. I am informed and believe that the 1999 publication date of this article that is after the October 16, 1998 filing date of the parent of this application, which issued as U.S. Patent No. 6,125,391. From my article, at p. 108:

> "Conceived originally as a CORBA-based interoperability framework, the eCo System architecture was recast in 1997 on an XML foundation, due to XML's simplicity and widespread adoption by key vendors, including IBM, Microsoft, Netscape and Sun."

On p. 109:

> "Business services in eCo were originally defined as CORBA application programming interfaces (APIs). While the CORBA approach appears workable within organizations that control APIs, our experience in several prototypes suggests that it is not practical for interenterprise integration. Fortunately, XML offers a promising alternative – agents interacting with business services through business documents."

Advantages of XML and a document interface, instead of a CORBA object interface, are explained by the article. This article confirms my recollection that we shifted from CORBA to XML after the May 1997 article was written, because CORBA proved impractical.

21. The process of preparing this declaration included meeting with Ernie Beffel and discussing the subject matter, asking him to prepare a draft declaration, reviewing and commenting on the draft, leading to this declaration.

I declare under penalty of perjury of the laws of the United States of America that the foregoing is true and correct. I make this declaration with the understanding and knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 U.S.C. and that making willful false statements would jeopardize the validity of my application and any patents issuing thereon.

Executed this __th day of March, 2006 in Palo Alto, California.

_____

Jay M. Tenenbaum, Ph.D.

On p. 109:

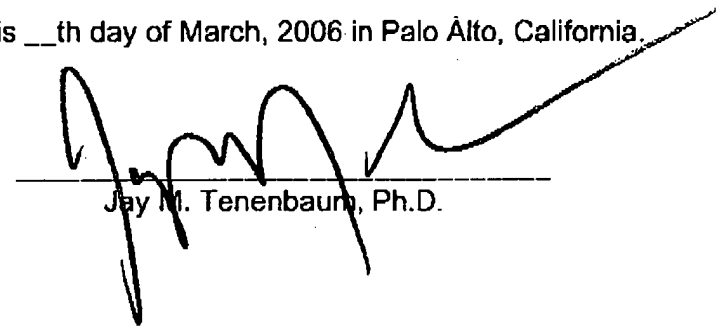
> "Business services in eCo were originally defined as CORBA application programming interfaces (APIs). While the CORBA approach appears workable within organizations that control APIs, our experience in several prototypes suggests that it is not practical for interenterprise integration. Fortunately, XML offers a promising alternative – agents interacting with business services through business documents."

Advantages of XML and a document interface, instead of a CORBA object interface, are explained by the article. This article confirms my recollection that we shifted from CORBA to XML after the May 1997 article was written, because CORBA proved impractical.

21. The process of preparing this declaration included meeting with Ernie Beffel and discussing the subject matter, asking him to prepare a draft declaration, reviewing and commenting on the draft, leading to this declaration.

I declare under penalty of perjury of the laws of the United States of America that the foregoing is true and correct. I make this declaration with the understanding and knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 U.S.C. and that making willful false statements would jeopardize the validity of my application and any patents issuing thereon.

Executed this __th day of March, 2006 in Palo Alto, California.



Jay M. Tenenbaum, Ph.D.

Curriculum Vitae
Jay M. Tenenbaum
December, 2003

Contact:
25 Alhambra Ct,
Portola Valley, CA 94028
650.851.8608 (home), 650.799.1767 (cell)
jmt@commerce.net

Personal: Born June 17, 1943. New York, NY, Married, one grown child
Citizenship: USA
Education:. BS (1964) and MS (1966), Elect. Eng, MIT; Ph.D. (1971) EE and CS, Stanford Univ.
Ph.D. dissertation: "Accommodation in Computer Vision"

Areas of Interest:
AI, Ecommerce, Healthcare Informatics
Specializations: Machine perception, AI applications in engineering and manufacturing, emarketplaces, business web services, Internet integration frameworks

Professional experience:

SRI: program manager for machine perception (1972-1980);

Schlumberger Palo Alto Research: Director, Artificial Intelligence Laboratory (1980-1990);

Enterprise Integration Technologies: founder and CEO (1990-1995); acquired by Verifone

Verifone: Vice President strategic technology (1995-1996),

CommerceNet: Chairman (1994-) and CEO (1996-1997, 2002-2003)

Veo Systems: Chairman and Chief Scientist (1998), acquired by Commerce One

Commerce One, Inc: Sr. Vice President and Chief Scientist (1999-2001)

Medstory, Executive Vice President and Director (2001 – present)

Webify Solutions Inc: Chief Scientist and Director (2002-present)

Consulting Professor of Computer Science, Stanford University (1986-1998)

Consulting Professor of Information Technology and Co-director, Center for Computational Health Sciences, Carnegie Mellon University (2003-)

Accomplishments:

Machine Perception - Theories of Intrinsic Images (1978), Interpretation-guided segmentation (1976), Role of perceptual organization in vision (1983); built and managed leading AI research groups at SRI (1970's) and Schlumberger (1980's)

AI in engineering and manufacturing: First-Cut (1987) and Next-Cut (1988) systems for concurrent product and process design; Manufacturing Knowledge System (MKS) (1989) for semiconductor manufacturing; PACT integration framework for collaborative engineering (1993).

Ecommerce - Pioneered commercial use of the Internet (founded first ecommerce company (1990), first Internet transaction (1992), first Internet auction (1993), founded CommerceNet - first ecommerce industry association (1994), first ebusiness web

services framework (CommerceNet's eCo System, 1996), first use of XML for ebusiness integration (1997), first industry exchanges (1999).

The above research was documented in refereed publications (Proc IEEE, AI Journal, IEEE Computer, CACM, CVGIP, IJCAI and AAAI, etc.). Ecommerce developments were also widely disseminated through keynote addresses at major industry events, as well as feature articles in trade, business and general publications including Business Week, Fortune, Time, Byte, Business 2.0, PC Week, Internet World, Networking. Computer World, and Info World; The eCo System Framework was the cover article in IEEE computer, May 97 as well as the subject of a technical paper in CACM in 1999..

Healthcare Informatics – Applying lessons from ebusiness to ehealth, specifically to address issues such as interoperability, integration and security that have impeded the development of National healthcare networks.

Proposed and architected an Internet clinical trials infrastructure for the NCI, as a member of Rick Klausner's Long Range Planning committee (1999, documented in the book Cancer Informatics); Built and deployed an open Web services-based network for real time processing of HIPPA-compliant insurance claims (2002), live in three states; Currently developing HealthConnect, a Community Health Network for Silicon Valley providers, payers, patients, pharmacies, labs, and government agencies. Stakeholders will be able to freely exchange clinical and administrative data, and subscribe to a diverse set of offerings ranging from credentialing to case management to knowledge-based medicine services. CommerceNet is developing HealthConnect as a regional pilot for the National Healthcare Information Infrastructure.

Affliations:

American Association for Artificial Intelligence (AAAI) – Chair, Conference Committee (1980 - 1988) and Board Member (1985-1988); Air Force Scientific Advisory Board (approx 1983-86); Founder and Chairman, CommerceNet (leading ecommerce industry association, 1994-Present); Advisory Committee, Center for Advanced Medical Informatics, Stanford University, Palo Alto, CA, (~ 1994-1996); National Cancer Institute, Long Range Planning Committee (1998-2001), IBM Institute of Advanced Commerce – Board (1998-99)

Corporate boards: Commerce One (1999-2001); Perfect Commerce (2000-present); Lumicyte (2001-present); Medstory (2001-present), Webify (2002-present)

Honors:

Fellow, American Association for Artificial Intelligence; Best paper award for work on Collaborative Engineering (1994); numerous invited international keynote addresses including Interop Japan (1997), First Internet World (1994), National Computer Conference of India (Bangalore, 1996), MITI symposium on ecommerce (Tokyo, 1997), Venture One (1998). (Note:all dates approximate.)

Selected Publications

Computational Vision:

"The Stanford Hand-Eye Project," with others, IJCAI 1969, 521-526

"An Accommodating Edge Follower," with K. Pingle, IJCAI 1971, 1-7

"The Use of Vision and Manipulation to Solve the Instant Insanity Puzzle," with others, IJCAI 1971, 359-364

"On locating objects by their distinguishing features in multisensory images," Computer Graphics and Image Processing, 2:308--320, 1973.

"On the automatic generation of programs for locating objects in the office scene," with T.D. Garvey, pages 162--168, 1974

"MSYS: A System for Reasoning about Scenes," with H.G. Barrow, Technical Note 121, Artificial Intelligence Center, SRI International, April 1976

"Experiments in Interpretation-Guided Segmentation," with H.G. Barrow, Artificial Intelligence 8(3): 241-274 (1977)

"Parametric correspondence and chamfer matching: Two new techniques for image matching," with H.G. Barrow, R. C. Bolles, and H. C. Wolf. in Proceedings of the International Joint Conference on Artificial Intelligence, pages 659--663, 1977.

"Recovering Intrinsic Scene Characteristics From Images," with H.G. Barrow, in Computer Vision Systems, A.R. Hanson and E.M. Riseman, Eds., Academic Press, New York, 3-26, (1978)

"Map-Guided Interpretation of Remotely-Sensed Imagery," with H.G. Barrow, R.C. Bolles, M.A. Fischler, and H.C. Wolf, Pattern Recognition and Image Processing, pp. 610--617 (February 1979)

"Interpreting Line Drawings as Three-Dimensional Surfaces," with H.G. Barrow, Artificial Intelligence 17 (1-3): 75-116 (1981)

"Detection of Roads and Linear Structures in Low-Resolution Aerial Imagery Using a Multisource Knowledge Integration Technique," with M.A. Fischler and H.C. Wolf, Comp. Graph. and Image Proc., 15, 1981, 201-223.

"Computational vision," with H.G. Barrow, Proc. IEEE, 69(5):572-595.(1981)

"Scene modeling: a structural basis for image description," with M.A. Fischler and H.G. Barrow, in Rosenfeld, A., editor, Image Modeling, 371--389. Academic Press, New York (1981)

"What Is Perceptual Organization For?" with A.P. Witkin, IJCAI 1983, 1023-1026

"On the role of structure in vision" with A.P. Witkin, in Human and Machine Vision, J. Beck, B. Hope, and A. Rosenfeld, editors, pages 481--543. Academic Press, New York, NY, (1983); reprinted as "On perceptual organization". In From Pixels to Predicates, A. Pentland Ed., pages 149--169. Ablex Publishing Corporation, Norwood, New Jersey.

"Retrospective on Interpreting Line Drawings as Three Dimensional Surfaces," with H.G. Barrow, Artificial Intelligence 59, pp. 71-80, 1993

AI in Engineering and Manufacturing:

"PIES: An engineer's do-it-yourself knowledge system for interpretation of parametric test data," with Jeff Pan, AI Magazine, Vol. 7, No. 4, 62-71 (Fall 1986)

"A framework for Knowledge-Based Computer-Integrated Manufacturing," with Jeff Pan and Jay Glicksman, IEEE Trans. Semiconductor Manufacturing, Vol. 2, No 2, 33-46 (May 1989)

"First-Cut: A computational Framework for Rapid Prototyping and Team Design," Proc. AAAI Spring Symposium on AI and Manufacturing, Stanford CA, March 1989

"A methodology and computational framework for concurrent product and process design," with M.R. Cutkosky, Mechanism and Machine Theory, 23(5), 1990

"Toward an Intelligent Agent Framework for Enterprise Integration." with J. Pan, IEEE Trans. on Systems, Man and Cybernetics, Vol. 21, No. 6, (November/ December 1991)

"Next-Cut: A Second Generation Framework for Concurrent Engineering," with D.R. Brown and M.R. Cutkosky, in Computer Aided Cooperative Product Development, D. Sriram and R. Logcher, eds., Springer Verlag, 1991

"Toward a framework for concurrent design," with M.R. Cutkosky, International Journal of Systems Automation: Research and Applications, 1(3): 239--261, 1992.

"Towards a knowledge medium for collaborative product development," with T. Gruber and J. Weber, in J.S. Gero, editor, Proceedings of the Second International Conference on Artificial Intelligence in Design, Pittsburgh, PA, pages 413-432, 1992. Kluwer

"Working with Multiple Representations in a Concurrent Design System," with M.R. Cutkosky and D.R. Brown, ASME Transactions, Journal of Mechanical Design. Vol. 114, No. 3, September 1992. pp. 515-524.

"PACT: An Experiment in Integrating Concurrent Engineering Systems," with others, IEEE Computer 26(1): 28-37 (1993)

SHADE: Technology for knowledge-based collaborative engineering" with J. G. McGuire, D. R. Kuokka, J. C. Weber, T. R. Gruber, and G. R. Olsen, Concurrent Engineering: Research and Applications, 1(3), 1993.

"Integrating General Purpose Planners and Specialized Reasoners: Case Study of a Hybrid Planning Architecture," with S.Kambhampati, M. Cutkosky, and S. Lee, IEEE Trans. on Systems, Man and Cybernetics, Special issue on Planning, Scheduling and Control, Vol. 23, No. 6, November/December, 1993.

"Agile infrastructure for manufacturing systems (AIMS)," with H. Park and R. Dove, in Proceedings of Defense Manufacturing Conference, San Francisco CA (1993).

"Collaborative Engineering Based on Knowledge Sharing Agreements," with M. Cutkosky, G.R. Olsen, and T. Gruber, Proc. of the 1994 ASME Database Symposium, 1994

"SHARE: A Methodology and Environment for Collaborative product Development," with M. Cutkosky, L. Leifer and J. Glicksman, *The Int. J. of Intelligent and Cooperative Information System*, vol.3, no.2, pp.129-53, June, 1995.

"Madefast: Collaborative Engineering over the Internet," with M.R. Cutkosky and J. Glicksman, CACM 39(9): 78-87 (1996)

Ecommerce:

"CommerceNet: Spontaneous Electronic Commerce on the Internet," with others, COMPCON 1995: 38

"Eco System: An Internet Commerce Architecture," with T.S. Chowdhry and K. Hughes, IEEE Computer 30(5): 48-55 (1997)

"Web Information Systems and Electronic Commerce. CACM 41(7): 89-90 (1998)

"An XML Framework for Agent-Based E-Commerce," with Robert Glushko and Bart Meltzer,. CACM 42(3): 106-114 (1999)

Healthcare Informatics:

"Cancer Informatics: Lessons from the world of ebusiness," in Cancer Informatics - Essential Technologies for Clinical Trials, M. Ball, J.S. Silva, J.V. Douglas and C.G. Chute, Eds., Springer-Verlag (Jan 2002)

Cover Feature

Eco System: An Internet Commerce Architecture

Robust electronic commerce will require several proprietary systems to interoperate. CommerceNet is proposing a framework of frameworks that will bridge among conflicting platform requirements.

Jay M. Tenenbaum

Tripatinder S. Chowdhry

Kevin Hughes
CommerceNet

The Internet is revolutionizing commerce. It provides the first affordable and secure way to link people and computers spontaneously across organizational boundaries. This is spawning numerous innovative enterprises—virtual companies, markets, and trading communities.

But the Internet's potential is imperiled by the rising specter of digital anarchy: closed markets that cannot use each other's services; incompatible applications and frameworks that cannot interoperate or build upon each other; and an array of security and payment options that confuses consumers.

One solution to these problems is an object-oriented architectural framework for Internet commerce. Several major vendors of electronic-commerce solutions have announced proprietary versions of such a framework. The major platforms are

- IBM CommercePoint
- Microsoft Internet Commerce Framework
- Netscape ONE (Open Network Environment)
- Oracle NCA (Network Computing Architecture)
- Sun/Javasoft JECF (Java Electronic Commerce Framework).

Recently, four of these companies have agreed to support a common distributed object model based on CORBA IIOP (Common Object Request Broker Architecture Internet InterORB Protocol). Yet for commerce on the Internet to thrive, such systems must also interoperate at a business application level. (For more information see the "Major E-Commerce Platforms" sidebar.) A consumer or business using one framework should be able to shop for, purchase, and pay for goods and services offered on a different framework. This is currently not possible.

In response, CommerceNet is organizing Eco System, a cross-industry effort to build a framework of frameworks, involving both e-commerce vendors and end users. This project is challenging from a technical perspective because information technology is moving so fast that there's seldom time for even de facto standards to emerge. Instead, we must deal with de facto interoperation—making incompatible products already in the marketplace communicate. Our philosophy is simple: Protocols, formats, and the like should not hinder business.

The success of this process clearly depends on market leaders in each area participating actively on their respective task forces. Admittedly, in past battles for market dominance (such as in operating systems and desktop PCs), it was difficult to bring leading players to the table. For robust Internet commerce, however, interoperability is so fundamental that we have to turn the concept of openness on its head—it's not just publishing an API. Everyone's software has to work together because no single company can control what platform its customers will use.

OVERVIEW

As proposed, Eco System will consist of an extensible object-oriented framework (class libraries, application programming interfaces, and shared services) from which developers can assemble applications quickly from existing components. These applications could subsequently be reused in other applications.

We are also developing a Common Business Language (CBL) that lets application agents communicate using messages and objects that model communications in the real business world. A network services architecture (protocols, APIs, and data formats) will insulate application agents from each other and from platform dependencies, while facilitating their interoperation.

Functionally, Eco System fills three distinct roles. It is

0018-9162/97/$10.00 © 1997 IEEE

- a layer of middleware that facilitates agent interoperation through services such as authentication, billing, payment, and directories;
- an object-oriented development environment that encourages the reuse of e-commerce modules (even modules that represent the product line of an entire company); and
- an industry roadmap and interoperability example that promotes open standards and helps technology vendors communicate with end users about product features.

A framework of frameworks

In object-oriented parlance, a framework is an almost complete application that users can customize or extend to address particular needs. Eco System is a framework for building Internet markets. Specifically, it's a framework of frameworks that model key business processes and services. Because frameworks build on each other, the resulting applications are tightly linked through a shared-services infrastructure. Eco System's frameworks fall into four general categories, as Figure 1 shows.

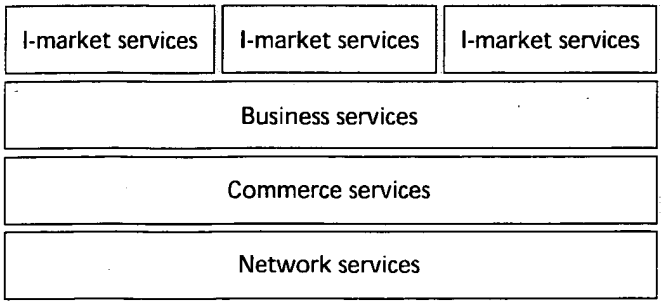


Figure 1. Four general categories of Eco System frameworks.

- I-*market services* are those that serve an Internet market. These are vertical markets of closely aligned businesses. Examples are real estate (title search, loan, and escrow services), securities trading (buy, sell, and quote services), or any vertical supply chain ("solicit bid," "issue request for quote," and "issue purchase order" services).
- Business services include generic business processes and applications common to multiple I-markets. These include retail (shopping, order fulfillment, and shipping) and business-to-business (procurement, order entry, inventory and supply chain management, and logistics) functions. Vendors may have initially developed such services for a specific I-market and later general-

Major E-Commerce Platforms

IBM's CommercePoint, a suite of e-commerce services, attempts to provide end-to-end business solutions (http://www.internet.ibm.com/commercepoint). It includes software packages for electronic storefronts (including credit card transactions using SET and back-office functions), purchasing (requests for proposals, electronic data interchange, and bidding), and distribution.

Netscape ONE (Open Network Environment) is a platform-independent, network-centric application development environment based on publicly defined open standards (http://home.netscape.com/comprod/one/white_paper.html). Key technologies include HTML, Java and JavaScript 1.1, CORBA IIOP, and broad support for open communication and collaboration protocols (HTTP, NNTP, SMTP, IMAP4, and POP3) and security services (Secure Sockets Layer 3.0 and X.509v3). Applications interact through these interfaces (available on Netscape clients and servers), eliminating the sharp distinction between client- and server-side development.

Oracle's Network Computing Architecture (NCA) combines Web technology (HTTP and HTML) with CORBA 2.0 and IIOP to provide distributed computing in a networked environment. NCA also supports ActiveX/COM clients through open COM/CORBA interoperability specifications ratified by the Object Management Group. Key components include "pluggable" objects called cartridges that use IDL to identify themselves to other objects in a distributed system (see http://www.oracle.com/nca/html/nca_wp.html).

Sun and JavaSoft's Java Electronic Commerce Framework (JECF) is an open platform for purchasing, banking, and finance (http://www.javasoft.com/products/commerce). It provides a user interface (or wallet) for online purchasing and other financial transactions; a secure, encrypted wallet database; access to strong cryptography; applets; and a purchasing infrastructure. Java Cassettes implement specific online transaction protocols such as SET, Mondex, and CyberCash CyberCoin.

These four vendors announced this March that they would redesign their networking products to support CORBA. Moreover, they promised to deliver some of these CORBA-compliant versions as early as this month. They are also expected to endorse the use of Java Beans, a platform-independent, component-based software architecture based on Java (see http://splash.javasoft.com/beans/WhitePaper.html).

This leaves Microsoft, which uses its proprietary Distributed Component Object Model (DCOM) architecture, as the major non-CORBA-compliant holdout. DCOM is an OLE derivative for networks, which runs only on Windows and also uses Microsoft's proprietary ActiveX components. These technologies support Merchant Server, a Microsoft product that allows Internet service providers to offer electronic storefronts supporting SET for about $3,500 (see http://www.microsoft.com/merchant). Industry observers point out that Microsoft recently endorsed a Hewlett-Packard proposal to bridge the ActiveX and CORBA object models.

Although the companies supporting CORBA are CommerceNet members, Microsoft is not. This situation—in which the major market shareholder fails to participate—is common to similar industry consortium efforts. As CommerceNet's interoperability initiatives gain momentum, we hope that Microsoft will become an active participant.

Table 1. Sample service request messages.

Service	Message
Payment	Make a payment
	Obtain payment
	Use a credit card
	Have I been paid yet?
Shipping	Schedule a shipment
	Check the status
	Get a quote
Catalog	Perform a search
	Add, delete, or modify listing

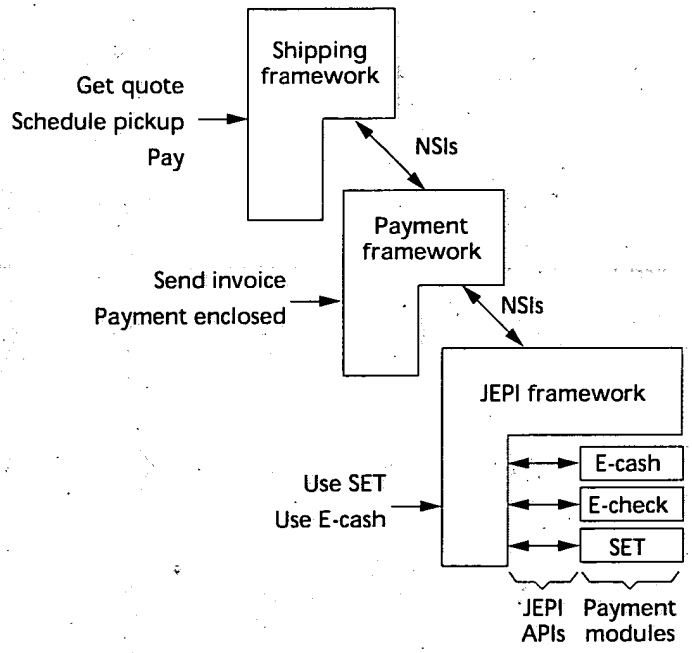


Figure 2. Frameworks communicate among themselves via NSIs and with application modules via APIs.

ized them for reuse. Marketware is a special subclass of these services that links buyers to sellers. (See the "Marketware" sidebar.)

- *Commerce services* are basic e-commerce services, such as digital "wallets," that allow individuals and companies to authenticate their identities, make payments, locate vendors, collaborate, and otherwise participate in an I-market. Advanced, next-generation commerce services will include secure multimedia mail, smart-card-based security and payment, digital-content delivery, application billing and accounting, transaction management, and agent management.
- *Network services* enhance the performance, reliability, and security of the Internet to accommodate mission-critical business needs. Examples include , quality-of-service management, IP (Internet Protocol) multicast, delivery receipts, authenticated packets, and smart firewalls (those that pass packets only among authorized business partners).

Each framework specifies core services that all application objects belonging to that class (for example, payments and catalogs) must provide. They must also specify a network services interface (NSI). An NSI is a set of messages in an implementation-independent language (CORBA IDL, Interface Definition Language). These standard messages request services over a network and differ from APIs in that they are at a higher level and written in IDL. In addition, a framework must specify APIs for software modules involved in delivering services.

Services

Every application under Eco System—whether a catalog or an entire I-market—is a network-accessible service. Table 1 illustrates a few core services provided by three representative frameworks. The table lists paraphrasings of the NSI messages used to request the core services. These core services literally define what it means to be, for example, a payment, shipping, or catalog service. Vendors will differentiate their products by providing additional services beyond those specified in the framework. But the defining characteristic of a payment, shipping, or catalog object is its ability to respond to the minimal set of core service requests specified in the associated framework.

Modules can plug into frameworks via APIs; thus, some frameworks function as middleware, allowing access to several vendors' modules through a common set of requests. Object wrappers transform standalone and legacy applications (written before a relevant Eco service framework existed) into Eco services. Application modules plug into e-commerce platforms via APIs, and other applications can access them using standard NSI requests. The JEPI framework is an example of a payment platform. When fully developed, it will define standard APIs and protocols that allow interoperability of many incompatible payment solutions already on the market.

Figure 2 illustrates the hierarchical relationship of frameworks and the roles of NSIs and APIs.

GETTING FRAMEWORKS TO TALK

We are basing Eco System on CORBA 2.0, an emerging industry standard for distributed objects and networking. CORBA 2.0 includes the Internet InterORB Protocol (IIOP), which Netscape Communicator will support. Eco will also work with HTTP (hypertext transfer protocol), HTML (hypertext markup language), and Java. Figure 3 shows the Web-based architecture.

The following design decisions conform to emerging industry trends:

Marketware

A special class of Eco applications and services brings together buyers and sellers. Marketware is based on a common platform that developers can customize by plugging in different application modules. These modules serve as building blocks to implement a variety of value-added markets and market services:

- *Matchmaking* is a trading post where buyers and sellers can exchange goods or services. This service matches buyers and sellers on the basis of product descriptions and personal or company profiles (like, for example, Sun's Matchmaker).
- *Negotiation* services allow buyers and sellers to post offers specifying price ranges, quantities, delivery dates, and other terms. The service notifies parties in real time or via e-mail of close matches. Parties can respond by modifying their offers if so desired (as in, for example, the FastParts system).
- *Buy-sell brokering* allows buyers to post requests for quotations, which the service forwards to registered sellers with appropriate interest profiles. Sellers can respond with bids, which the service collects, sorts, and forwards to the buyer. (Shopping agents such as Andersen Consulting's BargainFinder are a special case of this service.)
- *Referrals and directory* services handle buyer requests for referrals. These services match requests against profiles of registered sellers using buyer-supplied criteria.
- *Aggregation* allows buyers to submit requests for goods and services, which the service pools with similar requests to obtain quantity discounts.

The marketware framework supports these applications by providing a common set of structures and functions.

- Standard profiles for buyers, sellers, and intermediaries. Profiles provide the information needed for a party to participate in market transactions. This information could include size and type of business, location and street address, terms, conditions, contracts supported, certificate information, credentials, credit rating, and references.
- Standard taxonomies of goods and services would allow parties to target particular transactions and filter out others. Taxonomies would use standard commercial classifications such as SIC (standard industrial classification) codes as well as custom ones. For example, a three-level hierarchy would classify products by industry (for example, computer), subarea (peripherals), and type (disk drives). CommerceNet is working to develop an evolvable "Taxonomy of Everything" for products.
- Standard CBL commands to invoke market actions such as buy, sell, bid, post request for quote, and locate interested buyers or qualified vendors.
- Authentication and authorization functions that use buyer and seller profiles to control what information a party can see or modify.
- Accounting and reporting of transactions for buyers, sellers, and market administrators.
- A notification service allows buyers and sellers to register their interest in selected market events (a new-bid posting, for example) and receive a CBL notification message when they occur.

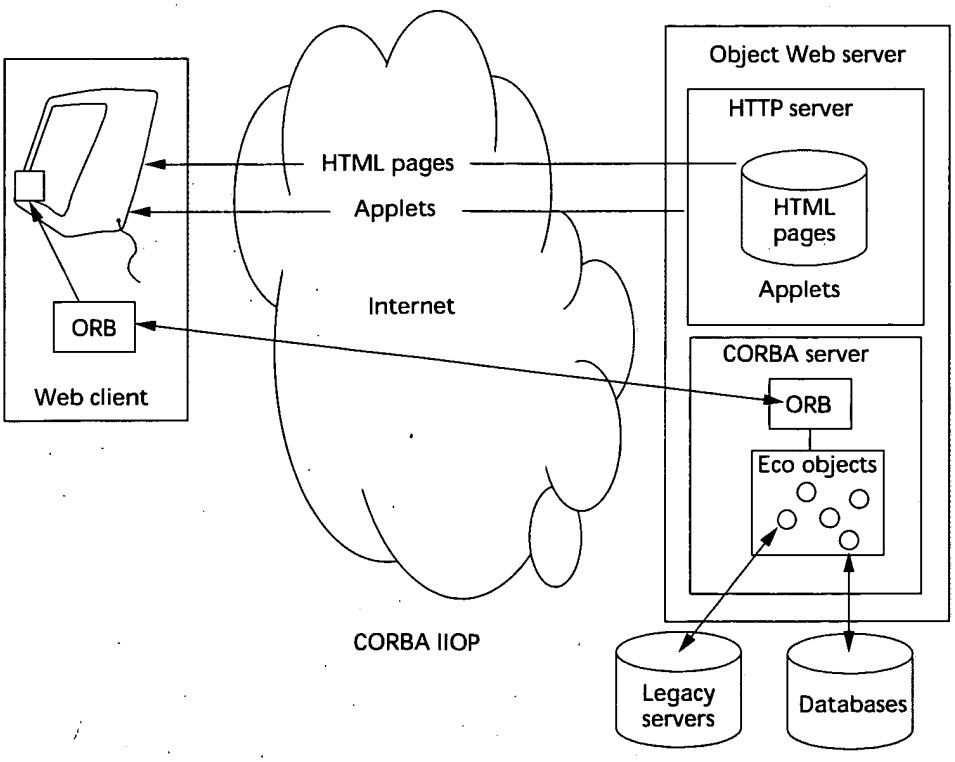


Figure 3. Eco services will be available as objects accessible via CBL commands sent over IIOP or HTTP/HTML sent by a browser. The architecture also incorporates Java applets, which link Web services to more robust transaction-oriented services via IIOP.

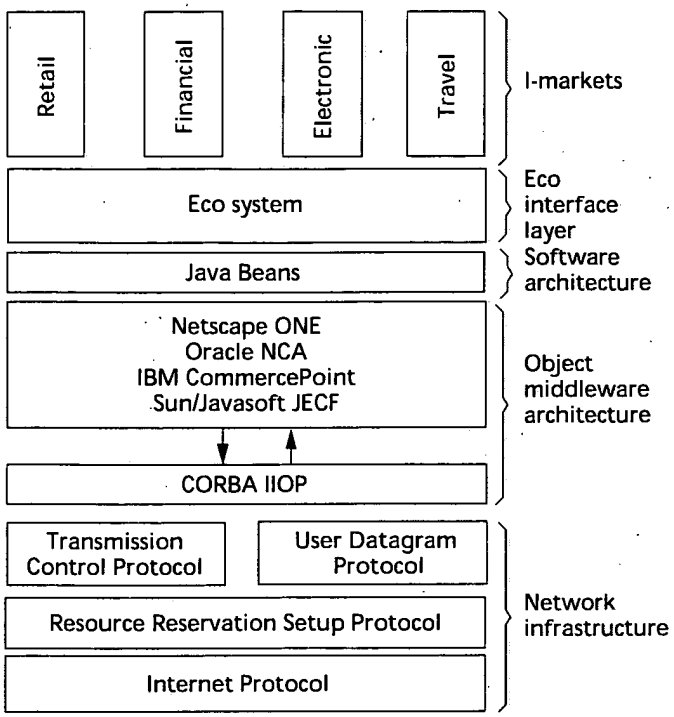


Figure 4. Protocol stack.

- *Network services.* Every Eco application will be a network-accessible service provided by agents.
- *Object Web.* Eco agents respond to CBL messages from other agents and to HTTP requests from browsers.
- *Industry compatibility.* As currently planned, Eco will foster interoperability among four of the five major e-commerce frameworks.
- *De facto interoperation.* Eco focuses on interoperation rather than standards. It will achieve interoperation in many ways, including the use of de facto standards implemented in Java and IIOP to achieve platform independence. Protocol negotiation, gateways, and mediators will provide semantic interoperation.
- *Scaleable, interchangeable building blocks.* Agents can direct CBL commands to a business, several businesses that have linked their catalogs or processes, a market (comprised of many companies), or a third-party intermediary.
- *Transparent outsourcing.* Eco will facilitate the outsourcing of business processes such as fulfillment, shipping, and payment processing.

Object orientation

Every Eco System service is a network-accessible object. As shown in Figure 3, objects respond to agents using CBL commands delivered over IIOP and to browsers using HTTP, HTML, and Java. This duality maintains compatibility with current Web sites and affords a graceful migration path. It's also compatible with emerging industry trends and anticipates the possibility that the next generation of HTTP and IIOP may someday merge. If the industry does not widely accept CORBA, agents will still be able to access the Web by using embedded semantic markup. Such embedded markup will let agents understand and respond to the information depicted graphically in a Web page. Microsoft and Netscape recently endorsed XML (Extended Markup Language), a simplified version of SGML used for embedding tags into HTML.

As shown in Figure 4, Eco imposes a layer of middleware on top of leading Internet commerce platforms such as Netscape ONE and Oracle NCA. It uses the CORBA IIOP architecture supported by these platforms and extends it to accommodate CBL agents.

Object bus. In CORBA, all objects connect to a common object bus, as shown in Figure 5. Thus, although we often depict Eco services hierarchically as in Figure 1, their actual implementation is flat; any Eco object can request a service from any other. This is convenient because situations do frequently arise where objects lower in the hierarchy require services from

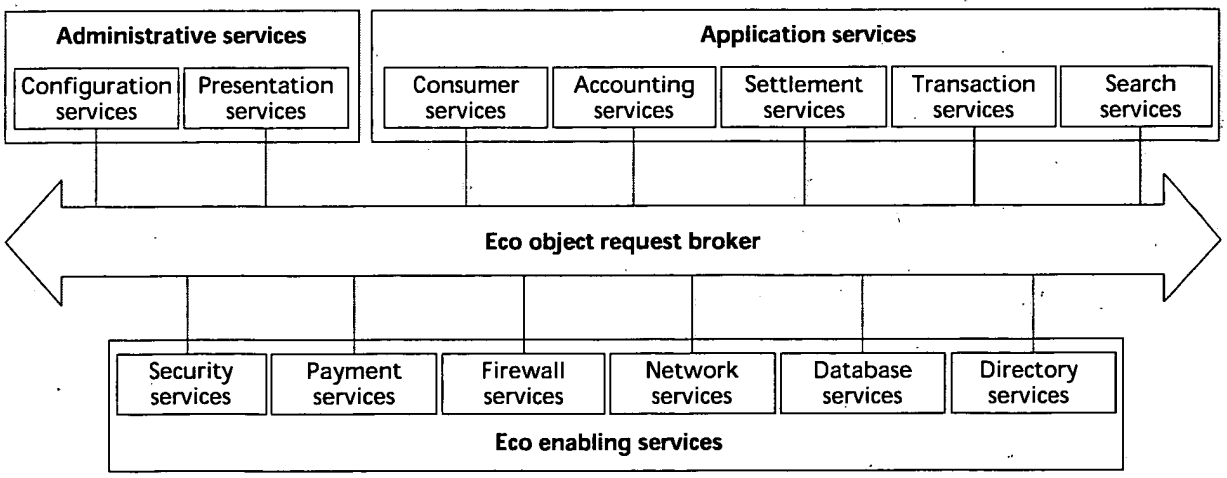


Figure 5. ECo object request broker acts as a bus between object-encapsulated services.

those above. For example, premium network services such as quality-of-service management or IP multicast may involve payments. Or a fulfillment service may need transportation I-market services.

IDL. CORBA and IIOP insulate application developers from most implementation and runtime details. CORBA provides IDL, a neutral definition language not tied to any specific programming language. Compiling the IDL generates object-oriented code implementing APIs. This allows any vendor to provide application object(s) that actually implement the specification. Vendors can write such objects in any language, and the objects can reside on any Internet-connected host. This architecture accommodates legacy applications by encapsulating them in an object wrapper and creating a corresponding IDL file as an interface. CORBA standardizes a CORBA-IDL-to-C++ mapping. JavaSoft and the OMG (Object Management Group) have released Java IDL alpha 2.2 for mapping IDL to Java.

Java. Object orientation allows developers to more quickly write and/or reuse applications to support changing business environments. Maintaining Eco's object orientation requires the use of an object-oriented language; CommerceNet has selected Java for this task.

Java is an interpreted language developed specifically with heterogeneous distributed networks and applications in mind. Vendor-neutral bytecode can be securely downloaded from the network as an applet that runs on a virtual machine residing on the user's system, most likely a Java-enabled browser. The Java runtime has built-in security features such as a bytecode verifier that enforces the Java security model (for example, disallowing pointers) and prevents malicious code from escaping the Java virtual machine (or "sandbox") and accessing the underlying operating system. Finally, Java Beans provides an architecture and platform-neutral API for creating and using dynamic Java components. Developers will be able to use a variety of development tools to assemble custom applications. These applications can draw on a rich variety of support services (such as event handling and persistence) that make Java Beans fully portable.

Protocol negotiation, mediators, and gateways

Application vendors are usually much more willing to agree on a metaprotocol than a standard. That's because a standard would require most to abandon rival technologies in which they have a substantial investment. Since today's computers can support multiple protocols, negotiation is a practical way of realizing de facto interoperation.

Negotiation protocols, bridging gateways, and mediators (smart gateways) have a part in accomplishing interoperability. Often, an application may not care what protocol it uses: "Just tell me what protocol you prefer, and I'll accommodate it if I can." This is the basic philosophy underlying the JEPI payments framework (see the "Payment Inter-operability" sidebar). In JEPI, sellers provide buyers with a list of payment types they accept (analogous to merchants displaying credit card logos in their store windows). Buyers then select the form of payment they wish to use, which implicitly

Payment Interoperability

In December 1995, the World Wide Web Consortium (W3C) and CommerceNet cosponsored the Joint Electronic Payment Initiative (JEPI) to bring key industry players together (CyberCash, IBM, Microsoft, Xerox, and British Telecom, among others) to ensure that multiple payment instruments, protocols, and transports will interoperate over the Internet.

JEPI is a metaprotocol built on top of two new Web protocols—PEP (Protocol Extension Protocol) and UPP (Universal Payment Preamble)—that let clients and merchant servers negotiate among and select payment mechanisms. Clients and servers can ask each other what forms of payment they support and negotiate a mutually acceptable payment mechanism.

PEP is a protocol for extending HTTP so that it can dynamically deploy applications that require more facilities than those provided by HTTP's request-response model. PEP associates new extensions to HTTP with a URL and uses a new `Protocol:` header field to carry the extension identifier and other necessary information—including possibly an implementation of the extension—between clients and servers. Like Java's protocol handlers, PEP provides the capability to automatically and dynamically download software component interfaces, enabling sophisticated applications such as distributed authoring tools to interoperate over the Web. PEP has been submitted to the IETF for inclusion in HTTP.

Don Eastlake built the Universal Payment Preamble on top of PEP. UPP is intended to provide a minimal layer that lets customers use a multipayment wallet and easily move from payment to payment. It provides a uniform vocabulary and syntax for naming options common to many payment systems, enabling clients and servers to exchange the necessary information and enter a specific payment system. This approach redefines each proprietary payment system as an URL-identified, PEP protocol extension implemented by a generic UPP protocol and module.

UPP negotiations occur via exchange of PEP protocol headers before or during shopping. Negotiation requests available payment choices, presents multiple choices, demands or makes a selection, and accepts or rejects choices. The payment protocol guarantees security, not UPP.

JEPI completed phase 1 in April 1997 with a demonstration at the Sixth International World Wide Web Conference of a JEPI implementation comprising two payment instruments, CyberCash and GCTech's GlobeID. W3C met with its members at that meeting to consider phase 2 strategies, which may include: validation/revision of UPP/PEP (JEPI used the August 1996 version of PEP, which was subsequently revised for consideration by IETF); incorporation of more payment systems (SET and micropayments), smart card integration; wallet and cash register APIs; and extension of HTML for micropayments. CommerceNet has committed to phase 2 development, according to CommerceNet's Jim Galvin, project manager for JEPI. For more information, see Eui-Suk Chung and Daniel Dardailler's "White Paper: Joint Electronic Payment Initiative (JEPI)," http:/www.w3.org/pub/WWW/Payments/white-paper.html.

AIMSNet

Ram Sriram

AIMSNet, a product of the Agile Infrastructure for Manufacturing Systems (AIMS) program, is a working example of an I-market in the making. Using AIMSNet, an intercompany network (using the Internet) links companies like Lockheed Martin and its suppliers, allowing multicompany project teams to exchange technical and business information, collaborate on design, post quotes and purchase orders, tender or accept bids, find potential suppliers and partners and track project milestones. More than 10 companies currently use AIMSNet, and dozens more are joining soon.

One of AIMSNet's powerful features is support for collaborative design. Its Multimedia Environment for Collaborative Engineering (MECE) is an online, shared notebook system developed by Lockheed Martin. This allows project team members to assemble and share information, such as design rationale and program decisions, in the form of text, audio, video, and screen snapshots. It also accommodates 3D design and manufacturing information by using VRML. VRML provides a 3D model independent of any specific CAD program. Team members use these tools to review information, collect comments, and make recommendations and changes. Current AIMSNet users are large programs within aerospace companies that develop complex systems such as satellites, rocket engines, missiles, and so on.

This effort, funded by the US Defense Advanced Research Projects Agency, has also developed templates to standardize transactions between companies. An important e-commerce concept, templates convey information between companies in a standard format easily accessed from anywhere through a HTML browser. Users can import information from legacy systems as well as through industry standard protocols. These templates also serve as simple front-end-to-remote databases that are network accessible.

Work is in progress to provide multitier supply chain coordination and facilities for evaluating and selecting suppliers. The coordination agent enables team members to track events critical to a project's success. The agent filters, sorts, prioritizes, and presents status information coming from various sources to project members based on their requirements. This helps project members manage the project from their own perspective. The supplier selection agent provides a mechanism for rapidly identifying key partners that can meet a project's multiple criteria. AIMSNet currently offers users a preliminary version of these services.

AIMSNet, an industrial commerce infrastructure, is currently piloted as an aerospace I-market but can be easily customized to several other I-markets including automotive, electronics, and construction.

Ram Sriram is AIMS Program Director for Lockheed Martin Missiles & Space. Contact him at sriram@aic.lockheed.com.

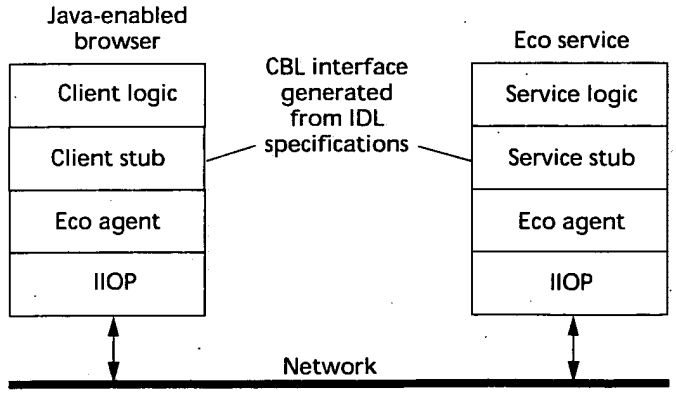


Figure 6. IDL provides a neutral definition language for connecting distributed applications (through CBL and Eco agents) in a platform-independent manner.

selects the appropriate protocol (SET or Mondex, for example).

An alternative to protocol negotiation is simply to translate between proprietary protocols using a gateway service. Gateways work well with functionally similar protocols that differ in syntactic details. Thus, gateways are often a good way for legacy database applications to communicate (for example, my SAP purchasing system can talk to your Oracle order-entry system) because the applications involved are reasonably well standardized at a functional level.

Gateways can also complement protocol negotiation. Namely, one alternative can be for each party to adhere to their favorite protocol and employ gateway services. In effect, the parties agree to disagree.

Mediators are smart gateways, which can negotiate a mutually acceptable protocol with each of several sites, retrieve information from each site, and integrate it. Mediators were originally developed for advanced information retrieval tasks, but are well-suited to e-commerce tasks such as integrating the catalogs and business systems of several cooperating firms.

Common Business Language

NSI messages, business objects, and product taxonomies will constitute a CBL for Internet commerce. Eco extends IIOP by adding two new levels of abstraction: CBL messages and CBL agents. A CBL message is an object-oriented alternative to the ad hoc text strings currently used in electronic data interchange. Each framework inherits the service requests and business objects of those frameworks upon which it builds, specializing and extending the inherited entities to provide new functions.

CBL agents provide a baseline set of common (Telescript-like) services that all e-commerce applications can build on. They include basic authentication,

EDI Interoperability Testing

EDIINT, the Internet Engineering Task Force (IETF) workgroup, has recommended standards for interoperable, secure electronic data interchange over the Internet. CommerceNet is sponsoring interoperability testing of these EDIINT recommendations among implementations from 10 vendors. These vendors include Actra Business Systems (a Netscape/GEIS company), Atlas Products International, AT&T, CyberPath, DaNet, Digital Equipment Corp., EDS, Harbinger, Premenos, and Sterling Commerce.

The vendors are checking the ability of their products to pass EDI securely among themselves. In January, five vendors demonstrated the successful exchange of digitally signed data among their products. This demonstration involved passing electronic documents over SMTP (Simple Mail Transport Protocol) using S/MIME (Secure Multipurpose Internet Mail Extension) encoding. Although the products all implement the S/MIME standard, factors such as certificate version differences and S/MIME support for multipart/signed documents still caused short-term interoperability problems.

Rik Drummond is chair of the IEFT working group, manager of the testing, and principal of the Drummond Group, an e-commerce consultancy. He anticipates the results of the EDIINT recommendations and the assurance of the CommerceNet interoperability testing to result in several secure, interoperable, off-the-shelf Internet EDI products in the next few months. The first group of five vendors will complete the total testing—exchanges of certificates, encrypted and signed messages, and signed receipts—by mid-May. A signed receipt is the basic mechanism for nonrepudiaton of receipt. In addition, the IETF is reviewing two draft standards—"MIME-Based Secure EDI" and "EDIINT Functional Specifications"—that outline the basis for secure, interoperable, Internet EDI. These standards set forth functional requirements for encryption, key management, content integrity, authentication, receipts, and tracking and error handling. They also recommend existing standards that fulfill these requirements. Both documents are available at http://www.ietf.org/ids.by.wg/ediint.html. Drummond expects these drafts to be accepted as Requests for Comments within the next few months. For more information about either the CommerceNet interoperability testing or the standards, contact him at drummond@onramp.net.

authorization, billing and accounting, micropayment, and directory services. We will base these agents on several lightweight agent architectures developed for use with Java, including IBM's Aglets and Mitsubishi's Concordia.

Eco's agent platform, depicted in Figure 6, provides an agent transport protocol and associated management and support services (creating and destroying agents, subcontracting tasks, delegating permissions and resources, and administering offers to buy or sell services). Using IDL, the CBL stub translates CBL messages into object requests to use IIOP-provided interoperability services.

PROJECT STATUS

In addition to the four major platform vendors, other organizations are active CommerceNet participants—Actra, Bank of America, Visigenic, the World Wide Web Consortium, and NIST, to name a few. CommerceNet recently agreed to cooperate with five Japanese organizations—NTT, the Japan Research Institute, Mitsubishi, NEC, and Oki—in developing functional prototypes of I-markets for a mall of malls and auto parts procurement. Additional I-market pilot programs include those for real estate and aerospace. The latter is already a working Internet-based network for manufacturing procurement; see the "AIMSNet" sidebar.

Another area in which CommerceNet is making a significant impact is in establishing standards and testing for secure electronic data interchange (see the "EDI Interoperability Testing" sidebar).

Although projects like AIMSNet allow pre-established trading partners to work together, we will use its results and EDI to create open I-markets in which an entire industry can come together for trade.

Internet commerce stands at a critical juncture. After an exhilarating start-up, further development hinges on bridging the chasm between early adopters and a true mass market. We envision Eco System as the foundation of that bridge.

Eco System is not just about creating an architectural framework of frameworks. It is, more importantly, about establishing an ongoing process and organization for achieving broad industry consensus on interoperability and reuse issues critical to open e-commerce. These issues are changing daily; visit http://www.commerce.net/Eco for the latest information. ❖

Jay M. Tenenbaum is founder and chair of CommerceNet, an industry association for e-commerce.

Tripatinder (Trip) S. Chowdhry, cofounder of the CommerceNet group, is the chief architect of Eco System. Chowdhry received his MBA from Kellogg Graduate School of Management and an MS in computer science from the University of Southern California.

Kevin Hughes is a consultant and CommerceNet Fellow.

Contact Tenenbaum at CommerceNet, 4005 Miranda Ave., Suite 175, Palo Alto, CA 94304; jmt@commerce.net.

ROBERT J. GLUSHKO, JAY M. TENENBAUM,
AND BART MELTZER

AN XML FRAMEWORK FOR Agent-based E-commerce

Emerging standards for commercial document exchange promise open business-to-business e-commerce.

COMMERCENET'S ECO SYSTEM INITIATIVE, LAUNCHED IN 1996, aims to transform the World-Wide Web into an agent-based infrastructure for Internet commerce.

Today's Web gives people unprecedented access to online information and services. But its information is delivered in format-oriented, handcrafted hypertext markup language (HTML), making it understandable only through human eyes. Software agents and search engines have difficulty using the information because it is not semantically encoded. Clever programmers work around some of HTML's inherent limitations by using proprietary tags or software that "scrapes" Web pages to extract content. Unfortunately, such ad hoc approaches do not scale. Proprietary tags require browser plug-ins, and scraping approaches require a customized script for each Web site. These approaches balkanize the Web, making it inaccessible to agents.

Tomorrow's Web will use the extensible markup language (XML) to encode information and services with meaningful structure and semantics that computers can readily understand. In Internet commerce, companies will use XML documents for publishing everything from product catalogs and airline schedules to stock reports and bank statements. They will also use XML forms to place orders, make reservations, and schedule shipments. Any agent with the proper authorization will be able to obtain computer-interpretable data sheets, price lists, and inventory reports through the Web or email, then request quotes, place orders, and track shipments.

By making the Web accessible to agents and other automated processes, XML will fundamentally transform the nature of e-commerce (see Maes et al.'s "Agents That Buy and Sell" in this issue). XML will eliminate the need for custom interfaces with every customer and supplier, allowing buyers to compare products across many vendors and catalog formats, and sellers to publish their catalog information once to reach many potential buyers. Online businesses will also be able to build on one another's published content and services to create innovative virtual companies, markets, and trading communities.

Web merchants might initially dread that XML-encoded information makes it too easy for buyers to compare prices and competitors to co-opt their content. But fear of lost business opportunity as e-commerce grows and the recognition that XML provides many other advantages for sellers (such as the ability to differentiate products in ways other than price) are likely to convince them to adopt richer markup formats. (see Wong et al.'s "Java-based Mobile Agents" in this issue). In time, most merchant Web sites will provide agent-searchable catalogs that supply product descriptions, as well as information about price and availability.

For consumers, the most obvious result of pervasive markup will be smart shopping agents that level the playing field in their dealings with sellers. Using Internet-wide shopping directories, these agents will be able to locate all merchants carrying a specific product or service, then query them in parallel to locate the best deals. Some merchants will provide sales agents that negotiate with shopping agents and generate customized offers in response to their solicitations. The shopping agents can then sort the offers they receive according to criteria set by their owners—the cheapest flight, the most convenient departure time, the roomiest aircraft, or some weighted combination. Cybermediaries will offer innovative brokering and referral services that match buying and selling agents, as well as order-aggregation services that increase their purchasing clout.

Agent-based shopping by consumers is just the tip of the e-commerce iceberg. Whenever a product is bought, information propagates back down the supply chain, triggering a series of distribution, manufacturing, and logistics events. Today much of this business-to-business information is exchanged through EDI messages. But traditional EDI is complex and expensive, because most messages travel over proprietary networks. Moreover, EDI's brittle syntax necessitates a custom integration solution between each pair of trading partners.

For these reasons, EDI transactions will increasingly take place over the Internet using an XML/EDI message format. Such messages will be more economical than traditional EDI messages, while being easier to validate and translate into the formats needed by applications at each end of the exchange [4]. This development will encourage businesses, including many that find traditional EDI too costly, to implement Web agents that respond to XML messages. This agent-based approach to enterprise integration is simpler and more open than traditional EDI, because it avoids the "pairwise tyranny" through which big companies impose proprietary message formats on small companies. More-

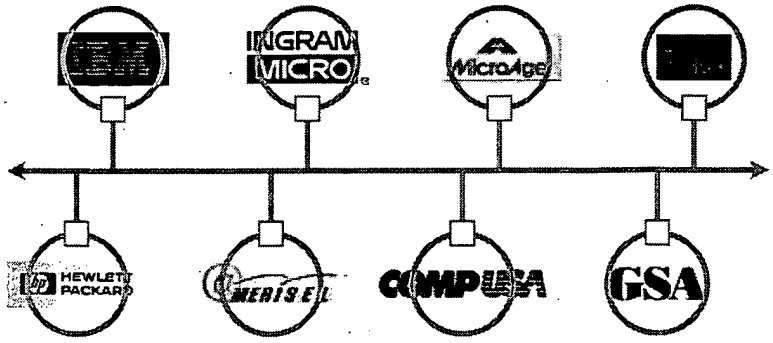


Figure 1. A supply Web linking PC manufacturers, distributors, and resellers

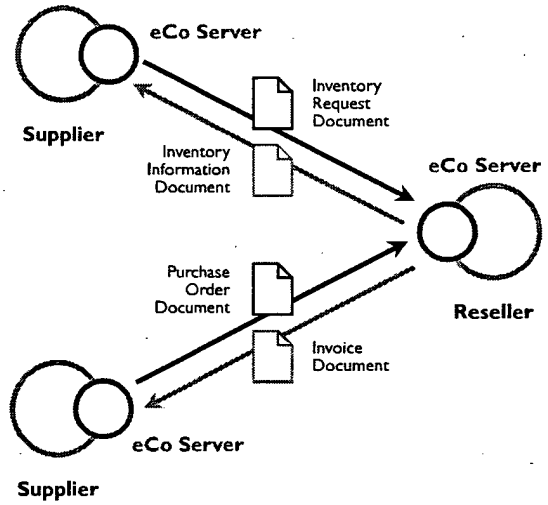


Figure 2. XML-based document exchange in the eCo System

over, publishing XML-encoded documents, such as data sheets and price lists, on the Web makes the information available instantly to all potential trading partners. Instant availability transforms rigid supply chains into "supply Webs," in which participants transact business spontaneously (see Figure 1).

The eCo System began as an architectural vision for open Internet commerce [5], proposed and evangelized by the 500-member worldwide CommerceNet Consortium in 1996. Conceived originally as a CORBA-based interoperability framework, the eCo System architecture was recast in 1997 on an XML foundation, due to XML's simplicity and widespread adoption by key vendors, including IBM, Microsoft, Netscape, and Sun.

Today's eCo System enables companies to communicate over the Internet using self-defining XML business documents that agents, as well as people, can easily understand. Business Interface Definitions (BIDs), posted on the Web, tell potential trading partners what online services a company offers and what documents to use when invoking those services. For example, a BID might allow a customer to order goods by submitting a purchase order or a supplier to check availability by downloading an inventory status report (see Figure 2).

A key element of the eCo System framework is the Common Business Library (CBL), an extensible, public collection of generic BIDs and document templates that companies can customize and assemble to go online quickly.[1] CBL includes XML message templates for the basic business forms used in ANSI X12 EDI transactions, as well as those used in such emerging Internet specifications as Open Trading Protocol (OTP) and Open Buying on the Internet (OBI). These specifications are mapped to each other using a dictionary of common business terms and data elements. A company can thus define its business interface in terms of any Internet standard mapped to CBL and communicate instantly with every other company that has done the same, even when the companies subscribe to different standards.

The eCo System framework overcomes two longstanding barriers to e-commerce. CBL facilitates spontaneous commerce between trading partners without custom integration or prior agreement on specific industrywide standards. And by being interpretable by both people and agents, XML documents provide an incremental path to business automation, whereby browser-based tasks are gradually transferred to computer agents. These advances eliminate much of the time, costs, and risks of traditional system integration. Moreover, the eCo System transforms closed trading partner networks into open markets and extends such enterprise applications as inventory management and production scheduling across entire supply chains.

XML is a simplified metalanguage, derived from SGML, emerging as the standard for self-describing data exchange in Internet applications. XML was developed by the World-Wide Web Consortium in 1997 and is being implemented rapidly by such major platform vendors as IBM, Microsoft, Netscape, and Sun Microsystems. XML's power

[1]The CBL was called the Common Business Language in earlier descriptions of eCo System. The change emphasizes CBL's function as a set of building blocks for XML applications and its role as a complement (rather than as a competitor) to ICE, OBI, OFX, OTP, RosettaNet, and other commerce languages.

derives from its extensibility and ubiquity. Anyone can invent new tags for particular subject areas, defining what they mean in document type definitions (DTDs). Content-oriented tagging enables a computer to understand the meaning of data, including, say, whether a number represents a price, a date, or a quantity.

This tagging significantly increases the functionality of Web e-commerce applications, because they can now do much more than simply display product data. For example, items in an XML-encoded catalog can be sorted by price, availability, and size.

One of eCo System's longstanding goals has been to enable businesses to build on one another's services to create virtual enterprises. Such plug-and-play commerce involves modeling enterprises as collections of services, some internal to a particular business, others provided by trading partners. Business services in eCo were originally defined as CORBA application programming interfaces (APIs). While the CORBA approach appears workable within organizations that control APIs, our experience in several prototypes suggests it is not practical for interenterprise integration. Fortunately, XML offers a promising alternative—agents interacting with business services through business documents.

Business documents represent a more intuitive and flexible way to access business services than programming APIs. It is much easier to interconnect companies in terms of the documents they exchange, on which they already largely agree, than in terms of their business system interfaces, which invariably differ. The coupling is looser, but loose coupling is better than no coupling at all.

XML's human readability is another significant advantage over CORBA. Just as HTML is a language for the eyes, CORBA is a language for CPUs, meant to convey information among programs, with no concession to human readability. XML documents are as readily interpretable by humans as they are by computers, especially with the aid of a style sheet [2].

Other proposals for agent languages suggest that first-order logic or other formal languages enable more precise specification of messages than XML [1, 3]. We prefer XML for two reasons—one language-theoretic, one practical. Expressing semantics in syntax rather than in first-order logic leads to a simpler evaluation function while needing no agreement on the associated ontologies. The practical argument, which is much more important for commercial success, is XML's ubiquity. The Web has made everyone appreciate the power of markup languages, practically assuring the widespread adoption of XML, as

Domain-specific Commerce Languages

The power of XML in enabling interoperability and simplifying the sharing and reuse of information between business domains is encouraging companies to work together to develop XML-based specifications for the business information they exchange most often. Sample specifications include:

- Open Trading Protocol. A consortium of banking, payment, and technology companies is specifying information requirements for payment, receipts, delivery, and customer support (www.otp.org). The goal of OTP is efficient exchange of information when the merchant, the payment handler, the deliverer of goods or services, and the provider of customer support are different entities with their own systems.
- XML/EDI. A group chartered jointly by CommerceNet, ANSI X12, and the Graphics Communication Association is defining how traditional X12 EDI business data elements should be represented using XML (www.xmledi.com).
- RosettaNet. This PC industry initiative is defining how to exchange PC product catalogs and transactions among manufacturers, distributors, and resellers (www.rosettanet.org).
- Open Buying on the Internet. The OBI initiative, launched by American Express and major buying and selling organizations, including Ford Motor and Office Depot, is automating large-scale corporate procurement of office and maintenance supplies (www.openbuy.org)
- Information and Content Exchange. CNET, News Corp., Vignette, and other information content providers are developing ways through ICE to create and manage networked relationships, such as syndicated publishing networks, Web superstores, and online reseller channels (www.w3.org/TR/1998/NOTE-ice-19981026).
- Open Financial Exchange. Originally proposed by CheckFree, Intuit, and Microsoft for the electronic exchange of financial statements among consumers, small businesses, and financial institutions, the OFX effort supports banking, bill payment, investment, and financial planning activities (www.ofx.net).

Share the Ontology in XML-based Trading Architectures

First bring semantic order to the world of XML.

Howard Smith and Kevin Poulter

Recent e-commerce application activity involving the extensible markup language (XML) has led to a proliferation of XML-based standards and markup language proposals. Among them are several designed to support site-to-site Web automation that lean naturally toward the agent paradigm of distributed computation.

Although XML represents a major step forward in e-commerce technology, business-to-business trading partners should also recognize XML's limitations. XML is not a cure-all for system interoperability, but a widely accepted foundation layer on which to build. Moreover, there are differing views on how to extend or complement XML to support agent-based e-commerce (see Glushko et al.'s "An XML Framework for Agent-based E-commerce" in this issue). This challenge is further complicated by debate over some fundamental questions: How should XML be extended to support the representation of business information? Should XML be enriched with tags reflecting higher-level concepts, especially business domains, such as standard business processes? How should foundation ontologies (from which higher-level content is composed) be defined? How can the numerous heterogeneous e-commerce frameworks (such as ICE, OBI, OTP, and XML/EDI) be unified to enable the expected low-friction market of the future? And will the future electronic marketplace be dominated by a series of commerce islands with trading groups isolated by the proprietary protocols and domain models with which their commerce agents interact?

Answers involve not only solving the related technology and intellectual challenges, but how to bring together the various communities of industrial standards developers. Each holds the essential elements of the overall solution. These communities, including EDI, Internet, knowledge engineering, and SGML, bring to the table subtly differing angles on the problem, including representation approaches associated with rich documents, publish/subscribe protocols, transactions, content syndication, and business semantics. To survive in this market, e-commerce component providers will have to support a number of different content formats and transaction frameworks, translating among them to achieve significant penetration. It appears that the main barrier to e-commerce lies in the need for applications to share information, not in the Internet's reliability and security.

Due to the wide range of enterprise and e-commerce systems being deployed by businesses and the way these systems are variously configured, the problem is particularly acute among large electronic trading groups. E-commerce will increasingly focus on trans-enterprise communication, while the number of trading partners and sophistication of e-commerce applications also increase. The need to unite business models, processes, and representation formats is greater than ever, while expectations run ever higher. Although many companies have already begun to organize, standardize, and stabilize their digital services in order to create and maintain sustainable network relationships with their trading partners, they are doing so only in conjunction with their immediate trading partners. This relatively narrow focus can limit the return on investment possible from each of these initiatives.

A global environment. There is now a need for e-commerce participants to create a global environment providing significant interoperability between the systems used by all engaged. Such an environment can be achieved through improved semantics within Internet transactions and in networked service definitions. It will facilitate consistent behavior among participants in large trading networks or within complex virtual organizations. Many of the foundation concepts needed to achieve this consistent behavior have already been established through work on distributed problem solving, intelligent agents, and knowledge sharing, yet to date these technologies have had little effect on Internet-based commerce.

Agent-based systems to support the next generation of Internet commerce must adopt common ontologies if they are to interact without misunderstanding. For example, content can be defined to enable application interoperation as well as information synthesis. An e-commerce standard being developed by major PC vendors, resellers, and distributors has shown by practical example in the PC distribution chain that quite sophisticated representation issues can complicate even straightforward commerce scenarios. For example, the required catalog model includes the need to represent the topology of the parts comprising a PC product.

But to bring semantic order to the world of XML, we have to be clear about what we mean by "ontology." The term is often used to refer to a vocabulary, yet even the terms within a simple vocabulary can be prone to misinterpretation, particularly in combination, unless they have been chosen carefully. Consider some of the problems already apparent in the plethora of e-commerce standards that

have emerged during the past few years. As new online trading environments are developed, the potential protocol mismatches between participants' commerce platforms can become major inhibitors to achieving industrywide e-commerce solutions and delivering supply-chain and market-efficiency benefits. Realizing Web automation in such complex environments reopens many of the problems and issues the knowledge-sharing and intelligent-agent communities have been wrestling with in such initiatives as the shared design environment, or SHADE, and the advanced technology operations system, or ATOS, using ontologies to enable agents working on different problems to interoperate over networks.

XML as a representation is just too forgiving at the document type definition (DTD) stage at the expense of the information processing stage. However, steps are being taken in the right direction; an example is the definition of schema languages to enable consistent schema semantics in the definition of objects in XML (such as by the World-Wide Web Consortium reflecting proposals from a number of organizations).

Consistent schema semantics will certainly enable efficient e-commerce using predefined DTDs between fixed networks of trading partners. But to enable the full benefits of agent-based e-commerce—where agents act in an autonomous or semiautonomous way, comparing and contrasting products or suppliers and negotiating with other agents—participating agents have to communicate in terms of a detailed ontology of the business domain.

The challenge for technology vendors, e-commerce participants, and standards bodies is to capitalize on the experience available in the knowledge representation and distributed agent communities.

Veo Systems is pursuing a pragmatic approach to solving some of these issues through the Common Business Library, an extensible, public collection of business interface definitions and document templates. This library is being rationalized and further developed by the CommerceNet eCo Framework Working Group established last year and should provide a foundation for addressing many of the unanswered questions in agent-based e-commerce. Ontologies will play a key role. C

HOWARD SMITH (howard.smith@ontology.org) is the director of Ontology.Org and a principal consultant (Internet) in Computer Sciences Corp. in Farnborough, Hampshire, U.K.
KEVIN POULTER (kevin.poulter@ontology.org) is chief technology officer of Ontology.Org and a principal consultant in Computer Sciences Corp., U.K.

HTML's heir apparent. XML may be theoretically less expressive than other formal languages, but we prefer a language that can be understood and produced by computer novices to a theoretically better one spoken only by computer scientists.

The significance of XML for integration extends beyond the Web to email, database records, and programming APIs. An XML parser imposes the same API on any XML data source, eliminating much of the need for custom programs to extract and integrate information from each source. So, integrating enterprise information from accounting, purchasing, manufacturing, shipping, and other functions can be accomplished by first converting each source to XML and then processing the parsed data stream. Put another way, each application need know only two source formats—its own and XML—rather than having to produce the native format of every other application.

XML by itself doesn't enable plug-and-play commerce. In addition to the language itself, a complete business integration solution also requires: standardized tags, or metadata, for each commerce community; a means for mapping between different metadata descriptions; and a server for processing XML documents and invoking appropriate applications and services. The eCo System framework starts with XML and adds these additional architectural and technology elements.

Specialized Markup Languages

XML makes it easy to create specialized markup languages that identify and describe buyers and sellers, the goods and services they want to buy or sell, and the various other document types involved in commerce. However, a vendor has obvious incentives for describing its offerings in ways that highlight its competitive advantages and that obscure comparison on features where it lacks an advantage. But if every business invented its own XML definitions for product catalogs, requests for quotes, price lists, purchase orders, invoices, transportation schedules, shipping notices, and delivery and payment receipts, the Web would be scarcely more usable as a platform for agents and other automated processes than it is today (see Smith's and Poulter's "The Role of Shared Ontology in XML-based Trading Architectures" in this issue).

Fortunately, many companies already recognize the need for information-exchange standards, uniting in several initiatives focusing on XML standards for particular industries or business

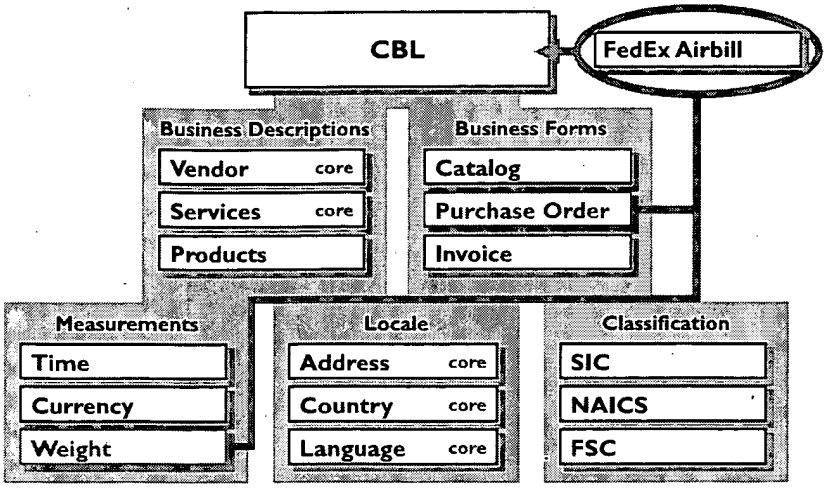


Figure 3. The Common Business Library

processes (see the sidebar "Domain-specific E-commerce Languages"). Unfortunately, these initiatives operate independently, doing little to facilitate interaction across industry and functional boundaries. The solution is to spur development of XML document models based on reusable semantic components common to many business domains. Such documents can be understood by any business through their common elements (such as address, date, and part number), while also providing a common mechanism for linking to the unique elements vendors need to differentiate themselves.

The CBL is designed to encourage development and use of generic XML document models. The library consists of information models for various concepts, including:

- Business descriptions, such as companies, services, and products;
- Business forms, such as catalogs, purchase orders, and invoices; and
- Standard measurements, such as date and time, location, and classification codes.

These models are represented as an extensible, public set of XML building blocks that companies can customize and assemble to develop XML applications quickly. Atomic CBL elements implement industry messaging standards and conventions, such as standard International Organization for Standardization (ISO) codes for countries, currencies, addresses, and time. Low-level CBL semantics are also derived through analysis of proposed metadata frameworks for Internet resources, such as the Dublin Core metadata element set developed by the Online Computer Library Center.

The next level of CBL elements use these building blocks to implement the basic business forms used in X12 EDI transactions, as well as those in OTP, OBI, and other emerging Internet standards.

A working group organized by CommerceNet and

```
<service>
<service.name>Order Service</service.name>
<service.location>www.veosystems.com/order</service.location>
<service.op>
<service.op.name>Submit Order</service.op.name>
<service.op.inputdoc>www.commerce.net/po.dtd</service.op.inputdoc>
<service.op.output>www.veosystems.com/invoice.dtd</service.op.outputdoc>
</service.op>
<service.op>
<service.op.name>Track Order</service.op.name>
<service.op.inputdoc>www.commerce.net/request.track.dtd<service.op.inputdoc>
<service.op.outputdoc>www.veosystems.com/response.track.dtd<service.op.outputdoc>
</service.op>
</service>
```

Figure 4. Fragment of an XML service definition for an eCo-compliant business application

other organizations recently began using CBL to create a base set of common terms, or mappings, between existing terms in commerce specifications, including OBI and OTP. The final result scheduled for release in mid-1999 will include a recommended base set of XML data elements, attributes, and definitions for use in e-commerce standards initiatives; they will be made freely available in public registries run by CommerceNet and other organizations. The Internet community, building on this foundation, will be encouraged to contribute additional elements and document models.

Agent-based shopping by consumers online is just the tip of the e-commerce iceberg.

Figure 3 shows how Federal Express might use CBL to create an XML version of its airbill by customizing a generic purchase order DTD with specific information about shipping weight. The generic purchase order, in turn, is assembled from more primitive CBL modules for address, date and time, currency, and vendor and product description. This example shows how reusing CBL components can significantly speed development of XML e-commerce applications and facilitate their interoperation.

When creating CBL, we found it helpful to extend XML with a schema language. The extensions add strong typing to XML elements so content can be readily validated. For example, an element called `CPU_clock_speed` can be defined as an integer with a set of valid values: {100, 133, 166, 200, 233, 266 Mhz}. The schema language also adds class-subclass hierarchies, so information is readily instantiated from class definitions. A laptop, for instance, can be described as a computer with additional tags for such features as display type and battery life. These and other extensions facilitate data entry, as well as automated translations between XML and traditional object-oriented and relational data models.

Trading partners not only have to agree on the meaning of message tags but understand how to use them for conducting business. In the eCo System, BIDs tell potential trading partners what online business services a company offers and which documents to use when invoking those services. In effect, services are defined by the documents they accept and produce. BIDs present a clean and stable interface to business partners, insulating them from a company's internal changes in technology, organization, and processes.

Figure 4 shows a fragment of a BID, defining an XML service for an eCo-compliant business. The service definition consists of two transactions—one for taking orders, one for tracking them. Each definition expresses a contract, or promise, to carry out a service if a valid request is submitted to the specified Web address. The order service requires an input document conforming to a standard `po.dtd` DTD in an industry registry operated by CommerceNet. If the service is able to fulfill the order, it returns a document conforming to a customized `invoice.dtd` whose definition is local. In effect, the company is promising to do business with anyone submitting a purchase order conforming to the XML specification it declares. No prior arrangement is needed.

A DTD is the formal specification, or grammar, for documents of a given type, describing the elements, their attributes, and the order in which they have to appear. For example, purchase orders typically include the names and addresses of the buyer and seller, a set of product descriptions, and associated terms and conditions, such as price and delivery dates. In the EDI world, the X12 850 specification is a commonly used model for purchase orders.

From Business Services to Virtual Enterprises

eCo servers provide the glue that links a set of internal and external business services to create a virtual enterprise or trading community. The server parses incoming documents and invokes the appropriate services (as specified by the applicable BID) by, say, handing off a request for product data to a catalog server or forwarding a purchase order to an enterprise resource planning system. The eCo server also handles translation tasks, mapping the information from one company's XML documents onto document formats used by its trading partners and into data formats required by its own legacy systems.

Following the service definition in Figure 4, when a company submits a purchase order, the XML parser in the eCo server uses the purchase order DTD `po.dtd` to transform the purchase order instance into a stream of information events. These events are then routed to any applications programmed to handle events of that type; in some

cases, the information is forwarded over the Internet to an entirely different business. In the purchase order example, information coming from the parser may be acted on by various applications:

- An order entry system processing the purchase order as a complete message;
- An enterprise resource planning system checking inventory for the products described in the purchase order;
- A customer database verifying or updating a customer's address;
- A shipping company system using the address information to schedule a delivery; and
- A bank system using credit card information to authorize a transaction.

However, what is most important in such processing is what is left out. Trading partners need agree only on the structure, content, and sequencing of the business documents they exchange, not on API details. How a document is processed and what actions result are strictly up to the business providing the service. This focus on commerce elevates enterprise integration from the system level to the business level.

A True Marketplace

eCo System's top-level goal is to transform the Web into a true marketplace by enabling spontaneous, peer-to-peer exchange of electronic business documents among all companies. This document-based approach replaces complex, expensive, and proprietary business integration solutions with one that is simple, affordable, and open.

The eCo architecture recognizes that a single dominant e-commerce standard is unlikely, even within a particular business community (and certainly not across communities). Rather, there will be many standards. CBL, in particular, is not a single standard but a collection of common business elements underlying all EDI and Internet commerce protocols. Its reusable components speed implementation of standards and facilitate interoperation by providing a common semantic framework. This approach to standards implementation and interoperation is fundamentally different from that taken historically by standards organizations and software vendors. It occupies an openness high ground embracing all the new competing standards being developed to take advantage of XML.

The eCo system framework and CBL are being evaluated in several of the standards initiatives listed in the sidebar on domain-specific commerce languages, as well as two major market trials sanctioned by CommerceNet:

- The U.S. General Services Agency (GSA). The largest buying organization in the U.S., GSA is creating catalog interoperability across numerous government agencies. Until now, the catalogs belonging to participating agencies were implemented as relational databases, as static files, or as catalog applications. An eCo server transforms each of these information sources into a standard catalog service that responds to CBL queries by outputting an XML data stream conforming to a common catalog schema. The integrated source catalogs can then be searched through specialized user interfaces developed by various participating technology vendors.
- RosettaNet. The RosettaNet consortium of PC manufacturers, resellers, and distributors is developing integration standards for the PC distribution channel; participants include Compaq Computer, CompUSA, Dell Computer, Hewlett-Packard, IBM, Ingram Micro, Merisel, Microsoft, and Tech Data.

The XML document models used in these initiatives are being rationalized to identify common semantic elements. These elements will be added to various public CBL repositories and made freely available (for more detail, visit www.commerce.net and www.veosystems.com). ■

References

1. Finin, T., Fritzson, R., McKay, D., and McEntire, R. KQML as an agent communication language. CIKM '94. In *Proceedings of the Third International Conference on Information and Knowledge Management*, 1994, pp. 456–463.
2. Fuchs, M. Domain-specific languages for ad hoc distributed applications. In *Proceedings of the Conference on Domain-Specific Languages*, 1997.
3. Kimbrough, S., and Moore, S. On automated message processing in electronic commerce and work support systems: Speech act theory and expressive felicity. *ACM Trans. Inf. Syst. 15*, 4 (Oct. 1997), 321–367.
4. Laplante, M. Making EDI accessible with XML. *EC.COM 4*, 2 (March 1998), 23–26.
5. Tenenbaum, J., Chowdhry, T., and Hughes, K. eCo System: An Internet commerce architecture. *Comput. 30*, 5 (May 1997), 48–55.

ROBERT J. GLUSHKO (glushko@veosystems.com) is director of information engineering at Veo Systems, Inc., in Mountain View, Calif.
JAY M. TENENBAUM (jmt@veosystems.com) is chairman and chief scientist of Veo Systems, Inc., in Mountain View, Calif.
BART MELTZER (bart@veosystems.com) is chief technology officer of Veo Systems, Inc., in Mountain View, Calif.